MEANT: Multimodal Encoder for Antecedent Information

Anonymous ACL submission

Abstract

The stock market provides a rich well of information that can be split across modalities, which makes it an ideal candidate for multimodal evaluation. Multimodal data plays an increasingly important role in the development of machine learning and has shown to positively impact performance. But information can do more than exist across modes— it can exist across time. How should we attend to temporal data that consists of multiple information types? This work introduces (i) the MEANT model, a Multimodal Encoder for Antecedent information and (ii) a new dataset called *TempStock*. *TempStock* consists of price, Tweets, and graphical data with over a million Tweets from all of the companies in the S&P 500 Index. We find that MEANT improves performance on existing baselines by over 15%, and that the textual information affects performance far more than the visual information on our time-dependent task from our ablation study. [1]

1 Introduction

Recently, multimodal models have garnered serious momentum, with the release of large pretrained architectures such as Microsoft's Kosmos-1 (Huang et al., 2023) and OpenAI's GPT-4 (OpenAI et al., 2023). Their general use has exploded in many different domains such as language and image processing (Lu et al., 2019; Kim et al., 2021; Huang et al., 2023). Particularly interesting to this study is the deployment of multimodal models on time-dependent environments such as the stock market. Recent successes have shown that event driven models processing multiple modalities are far more performant on stock market tasks than previously state of the art (SOTA) algorithms focusing purely on price information (Li et al., 2021; Zhang et al., 2022). Language data from news and social media sources have shown to greatly increase the performance of models for price prediction (Li et al., 2021; Zhang et al., 2022; Bybee et al., 2023; Mittermayer and Knolmayer, 2006; Xu and Cohen, 2018). However, these approaches typically lack attention components specifically designed to process inputs with sequential, time-dependent information (Li et al., 2021; Sun et al., 2017; Zhang et al., 2022; Xu and Cohen, 2018). This sort of data is particularly important when making predictions about stock prices or market movements, as price prediction is a time series task (Zhang et al., 2022; Xu and Cohen, 2018).

In this work, we introduce MEANT, a multimodal model architecture with a novel, temporally focused self-attention mechanism. We extract image features using a vision transformer architecture (Dosovitskiy et al., 2020) to find relationships in longer range information (i.e a graph of stock prices over a month), while extracting language features from social media information to pick up more immediate trends (e.g.: Tweets pertaining to stock prices over a 5 day period). Furthermore, we release *Tempstock*, a multimodal stock-market dataset that is designed to be sequentially processed in chunks of varying lag periods.

2 Related Work

Multimodal Models for Financial Twitter Data Several studies have employed natural language processing (NLP) techniques to financial markets, giving birth to the field of natural language-based financial forecasting (NLFF). Many of these studies have focused on public news (Ashtiani and Raahemi, 2023; Bybee et al., 2023). However, social media presents more time-sensitive information from active investors. Thus, for short term analysis, many researchers have begun to focus on Tweets for feature extraction (Araci, 2019; Wu et al., 2018), through which some have combined NLP techniques with traditional analysis on price

[1] The code and dataset will be made available upon publication.

data. Since Tweets often correspond to events as they happen in real time, such data is better suited for smaller windows (Xu and Cohen, 2018; Zhang et al., 2022). When working with stock market data, combining the features extracted through Natural Language Processing (NLP) methods with price data has shown promising results (Li et al., 2021; Zhang et al., 2022; Xu and Cohen, 2018). However, it is ineffective to feed the concatenated information to the model without encoding temporal dependencies (Li et al., 2021).

Modeling media-aware stock movements is essentially a binary classification problem. Many traditional machine learning methods have been deployed to solve it, including SVMs and Bayesian classifiers (Huang et al., 2012; Wang, 2003; Zuo et al., 2012). More recently, researchers have applied deep learning to the problem. Huang et al. (2016) used a convolutional neural network to explore the impact of Tweets on the stock market. Sun et al. (2017) and Selvin et al. (2017) then employed a recurrent architecture, specifically an LSTM, to extract relevant sentiments from Twitter data for stock market analysis, making their model multimodal, as it could handel Tweets and price information. Li et al. (2021) built atop this architecture, employing different tensor representations for their LSTM input to create more meaningful relationships between the price and Tweets data.

Xu and Cohen (2018) introduced StockNet, a large generative architecture built atop generative architectures, particularly the Variational Auto Encoder. StockNet represented the first deep generative model for stock market prediction (Xu and Cohen, 2018). TEANet, the most relevant work to our own, similarly used an LSTM to process their final output, but used a BERT-style transformer to extract relevant features from the Tweets (Zhang et al., 2022). TEANet is a language model equipped to handle lag periods similarly to MEANT. They concatenate their language features to price data as an input for an LSTM and a subsequent softmax temporal encoding. We abandon recurrence altogether, developing a novel temporal mechanism, entirely based upon traditional self-attention methods (Vaswani et al., 2017). The temporal processing in TEANet consists of concatenation methods similar to our own, but they do not employ attention over time. Furthermore, their model was built to handle Tweets and price inputs alone. MEANT can handle images as well, employing a dual encoder architecture similar to that of Su et al. (2023).

Financial Twitter Datasets Previous financial datasets have shown the power of Twitter data for financial analysis (Pei et al., 2022; Araci, 2019; Li et al., 2021). Twitter is powerful in its ability to generate real time information about the market before traditional newswires (Pei et al., 2022). Souza et al. (2015) focused on Twitter as a resource for examine financial dynamics in the retail sector. Pei et al. (2022) introduced TweetsFinSent, a large corpus specifically for sentiment analysis. Sun et al. (2017) introduced a dataset consisting of Tweets and prices, where the Tweets information served as a sentiment analysis accompaniment for the price data. Xu and Cohen (2018) introduced the StockNet-dataset, consisting of Tweets and price information for a selection of 88 companies over a two year period from 01/01/2014 to 01/01/2016. Mao et al. (2012) matched Tweets with price information from companies in the S&P 500 dataset, which is the most similar to the TempStock dataset that we introduce below.

3 TempStock Dataset

We collected a new dataset containing 1,755,998 Tweets and price information from all of the companies in the S&P 500 from 4/10/2022 to 4/10/2023.

From the price information, we calculated the Moving Average Convergence-Divergence (MACD) (Appel, 2005) for each company over a year. The MACD is built on the back of Exponential Moving Average (EMA) (Brown, 1964). The EMA is defined as follows:

$$EMA_t = (1 - \alpha) \cdot EMA_{t-1} + \alpha \cdot y_t$$

where t represents the day of EMA and y_t represents the closing price on that day, or in the case of the signal line, the MACD value on that day. α represents the degree of decrease; $\alpha = \frac{2}{t+1}$. Higher values, it can be observed, decrease more rapidly. The MACD consists of an MACD line, which is the difference between the fast EMA and the slow EMA (which are commonly set to 12 days and 26 days respectively), a signal line, which is the EMA of the MACD line itself (usally over a 9 day period) and a histogram, which is the difference between the MACD and the signal line. The MACD indicator was chosen[2] because it has been shown to perform well against other indicators in terms

[2]For more on this, see 6.4

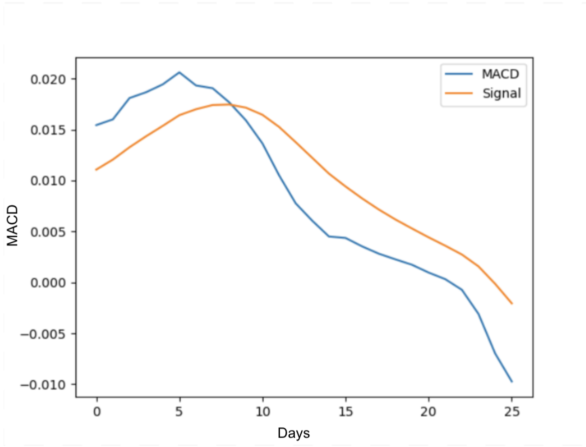

Figure 1: An example of a graph from our MACD data

of making accurate assertions about price directions (Appel, 2005; Chio, 2022). From our MACD data, we created graphs of the MACD indicator and the corresponding signal line over 26 day periods, which served as our image inputs to the MEANT model. A example of the graph inputs can be seen in Figure 1.

The MACD of each ticker in the subset was taken over a year period, along with the Tweets mentioning that company for each day in that period. The MACD information was gathered using the AlphaVantage api (Alpha Vantage Inc., 2024), and the Tweets were scraped using the snscraper (JustAnotherArchivist, 2021) in April 2023.

3.1 Preprocessing

First, all Tweets are anonymized, so that user identity is protected and potential noise in the dataset is reduced. Next, we created two different partitions of the TempStock dataset for pretraining and fine-tuning, called *TempStock-large* and *TempStock-small* as we wanted to have a partition of the data upon which to test the performance of the model. The total number of Tweets and MACD values can be found in Table 1.

TempStock-large is used for pretraining, contained Tweets, the MACD value, and the graphical representation for each ticker in the S&P 500.

TempStock-small contained a subset of the S&P 500, namely the first 37 tickers alphabetically. As we are tracing days where there was a recorded price, both the TempStock-small and TempStock-large dataset only trace weekdays, which amounts to 265 days in the aforementioned period. The number of Tweets for each ticker on each day varied,

Description	Count
Total Tweets	1,755,998
Total MACD Values	122,959

Table 1: TempStock-large Raw Numbers

as some companies were mentioned more often then others. TempStock-small required more direct preprocessing, as it was used for fine-tuning on downstream tasks. The raw data from TempStock-large was used for pretraining only.

In TempStock-small, Tweets, graphs, and MACD averages were arranged into 5 day lag periods, so that each data point processed by the model consisted of 5 MACD vectors, 5 days of Tweets, and a graph of the MACD indicator over the long period from each of those days (5 images containing graphs of the MACD indicator over 26 days leading up to said day). These data points were classified as *positive* if the below equation held for the target day (the last day in the lag period):

$$M_{t-1} < S_{t-1} \wedge M_t > S_t \wedge M_t > 0$$

The values are labeled as 1 (a buy signal, *positive*) if the MACD was above 0 on the target day and crossed the signal line, while experiencing an upwards trend in the succeeding week (higher lows). Otherwise they were labeled as 0 (*negative*). The totals for Tweets and MACDs can be seen in 2, along with the distribution of positive and negative buy signals.

Description	Total
Total Tweets	129,168
Total MACD Values	8,505
Positive MACDs	157
Negative MACDs	8,357

Table 2: Overview of TempStock-small

In TempStock-small, there was a class imbalance between *positive* and negative examples, which indicates that stocks to have sparse periods of momentum buy signals, according to the MACD ticker and traditional buy/sell strategies surrounding it (Joshi, 2022). For practical purposes, we would want a model that can accurately identify these sparse buy periods, and reject everything else. Thus, we employ the synthetic minority oversampling technique

(SMOTE) algorithm (Bowyer et al., 2011) to produce synthetic examples for our images, Tweets, and MACD price values. We clean our generated MACD values, to ensure that they obey our classification rules by a clear margin. In section 6.4 we discuss drawbacks and benefits of this approach. Furthermore, we generate our image and text data separately, to reduce noise between the two modality types. With our generated data, the class numbers change to the values in 3.

Category	Count
Positive	8,357
Negative	8,357

Table 3: TempStock-small Resampled

4 MEANT

MEANT combines the advantages of image and language processing with temporal attention, in order to extract dependencies from multimodal, sequential information, where 2 displays the full architecture. MEANT, similarly to most SOTA multimodal models (Liang et al., 2021; Kim et al., 2021; Su et al., 2019; Huang et al., 2023; OpenAI et al., 2023), is built atop the Transformer architecture (Vaswani et al., 2017).

4.1 Encoder Only

MEANT is an encoder-only model, similar to BERT (Devlin et al., 2018). The transformer stacks the attention mechanism with linear layers to extract relevant features from the input. Between the 2 parts of the encoder, and before the output, there is a standard residual connection, meaning that the input to that portion of the architecture is fed through added with the original input. This is done to alleviate the vanishing gradient problem (Pascanu et al., 2013). The encoder structure employed by both the language and vision pipelines is inspired by the Magneto model (Wang et al., 2022). It makes use of sub-layer normalization, meaning that a layer norm is interleaved between the attention and linear layer components of the encoder. This architecture was chosen because it has been shown to be successful on a wide variety of uni-modal and multimodal problems (Huang et al., 2023; Wang et al., 2022).

4.2 Token and Patch Embeddings

Before being fed to the attention mechanism, the two input types have to be prepared for processing using two different embedding strategies. The Tweets in MEANT are tokenized using the BERTweet tokenizer (Nguyen et al., 2020). MEANT also uses the BERTweet pretrained word embedding layer.

The images are first transformed into tensors of rgb values and reshaped to a manageable size. MEANT handles input image sizes of 4 x 224 x 224, where 4 represents the number of channels and the subsequent dimensions are the height and width respectively. These vectors are then broken down using the patch embedding strategy from the original vision transformer (Dosovitskiy et al., 2020).

4.3 Positional Encoding

In MEANT, the language and vision encoder use different variants of the rotary embedding (Su et al., 2021). The language encoder uses the $xPos$ embeddings (Sun et al., 2022), while the vision encoder uses 2D-axial rotary embeddings (Su et al., 2021), which simply means that the angle θ of rotation is altered according to the following equation:

$$\theta_i = i * floor(d/2) * pi$$

4.3.1 Temporal Attention

For the input to the Temporal attention mechanism, we used the pooled means from each modality, concatenating them to the MACD information from that 5 day lag period:

$$t_i = [t_p, g_p, m] \in \mathbb{R}^{l \times dim_t} \tag{1}$$

t_p is the mean of the Tweet language encoder outputs, g_p is the mean of the graph vision encoder outputs, l is the lag period, m are the MACD values, and dim_t is the temporal dimension, which is the sum of the language, image, and MACD dimensions. t_i signifies the input for the temporal encoder. In the vanilla implementation of the MEANT model, the temporal dimension is 1540. While many BERT-like architectures use [cls] tokens (Devlin et al., 2018; Araci, 2019), which are trained to become reasonable representations of the entire input over time, we found that mean pooling was a more effective strategy for performance from preliminary results.

In the case of MEANT, the outputs are not directly fed into a classification head, but are instead

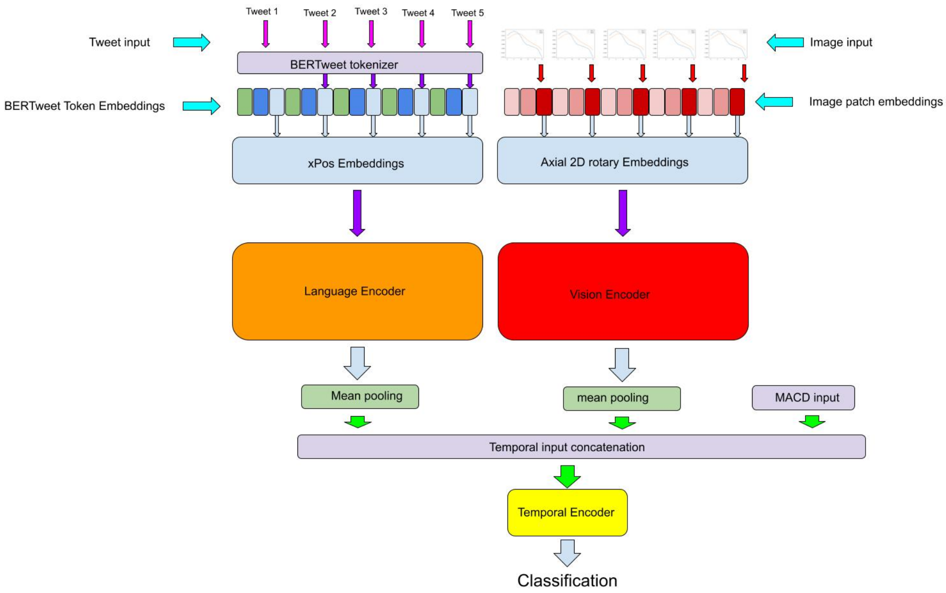

Figure 2: A schematic overview of the MEANT architecture

passed to a temporal mechanism. At this point in the pipeline, relevant image and text features have been extracted for each trading day in relation to themselves, not to one another. The temporal attention mechanism focuses on the target day, and its relationship to the preceding days. The purpose of the model is to extract a pattern from the preceding days, to identify future MACD crossings which may result in a profitable push. MEANT does this by using the query matrix in the attention mechanism, which only acts upon the target day, so that all of the keys and values are processed in relation to the target day.

$$tempAttention(Q, K, V) = softmax\left(\frac{Q_t K^T}{\sqrt{d}}\right) V$$

The relevant features are extracted by the language and vision encoders, where the temporal mechanism only needs to process a simple computation to find a meaningful temporal pattern. The temporal encoder is structured identically to the image and language encoders in all other aspects. There are positional temporal embeddings layered on top, but these are simply a learned parameter vector, not rotary embeddings. For the TempStock and Stocknet experiments (see 5 below), the output of the temporal encoder is then processed by the MLP head, which produces a classification.

5 Experiments

We ran the model at three different sizes, coined MEANT-small, MEANT-large and MEANT-XL. MEANT-small contained one encoder for language and vision, along with one temporal encoder. MEANT-large consisted of 12 encoders for both language and vision, with one encoder for temporal attention. 12 was selected as the number of encoders used in the original BERT model (Devlin et al., 2018). MEANT-XL had 24 encoders for the vision and language encoders. MEANT was implemented using a typical transformer formula, employing the use of RMSNorm (Zhang and Sennrich, 2019), Flash-attention (Dao et al., 2022), and GELU activation units (Hendrycks and Gimpel, 2016).

Model	**Parameter Count**
MEANT-small	73,685,762
MEANT-large	177,697,538
MEANT-XL	291,164,930

Table 4: MEANT Parameter Count

All fine-tuning and training was done with an AdamW optimizer (Loshchilov and Hutter, 2017), a cosine annealing learning rate scheduler with warm restarts (Loshchilov and Hutter, 2016), and an initial learning rate of $5e^{-5}$.

5.1 Pretraining

We follow typical pretraining methods for our language and vision encoders. For our language encoder, we used masked language modeling on our raw TempStock-large dataset. We trained our MEANT-small and MEANT-large language encoders on 4 NVIDIA p100 GPUs for 3 and 10 hours respectively. For MEANT-XL, we trained on an A100 GPU for 10 hours. A training batch size of 32 was used.

For the image encoders, we used masked image modeling with block and channel masking. The image encoders were trained on 4 NVIDIA p100 GPUs as well, for 20 hours. We used MACD graphs from the raw MACD data in TempStock-large. For these encoders, we also used a training-batch size of 32.

5.2 Fine-tuning on downstream tasks

We tested the viability of the MEANT architecture on two tasks.

5.2.1 TempStock

First, we wanted to see the performance of MEANT on TempStock-small. This boiled down to a binary classification task, identifying lag periods which resulted in momentum shifts and those that did not. We fine-tuned and tested the MEANT models on the augmented TempStock-small dataset, using a randomized split for our train, validation, and test data, consisting of 70%, 10%, and 20% of TempStock-small respectively.

To further measure MEANT's performance, we ran some similar SOTA encoder-based multimodal models on TempStock. TEANet, a key inspiration for this work, was the most similar model in original purpose, so proved the most interesting benchmark. We fine-tuned VL-BERT (Su et al., 2019) and ViLT (Kim et al., 2021) on TempStock-small as well.

5.2.2 Stocknet

The most similar dataset to TempStock was the Stocknet dataset (Xu and Cohen, 2018), which consists of Tweets and price values from a selected batch of stock tickers. Stocknet is different from TempStock as it is a unimodal dataset, containing no graphical component, and is furthermore focused on binary price change rather than momentum shift (as measured by MACD crossing in TempStock). Nonetheless, Stocknet represents one of the only datasets to our knowledge organized in lag periods and is therefore relevant as a benchmark for the MEANT model. StockNet, similar to TempStock, is a binary classification problem, where the inputs that had a movement ratio $\leq$ -0.5 were labeled 0 and the inputs with a movement ratio $\geq$ 0.55 were labeled with 1.

We ran MEANT-Tweets (both small and large) on the StockNet-dataset, and compared against TEANet (Zhang et al., 2022) which was originally evaluated by the authors on the StockNet dataset, as well as the StockNet model itself (Xu and Cohen, 2018). We ran a commonly used encoder architecture on the StockNet-dataset, fine-tuned with BERTweet (Nguyen et al., 2020). All experiments were ran for 10 epochs, and the results after the 10th epoch are described below.

Model	F1	P	R
VL-BERT	0.91	0.91	0.91
ViLT	0.94	0.95	0.94
TEANet	0.79	0.82	0.79
MEANT-base	0.97	0.97	0.97
MEANT-large	0.99	**0.99**	0.99
MEANT-XL	**0.99**	0.98	**0.99**

Table 5: TempStock Experiment Results, using Precision (P), Recall (R), and F-1 scores.

6 Results

Tables 5 and 6 in sections 6.1 and 6.2 show the results for our experiments respectively.

6.1 TempStock Experiment results

MEANT-base, MEANT-large and MEANT-XL outperform the similar models by a signficant margin. MEANT outperforms TEANet, the only other model with a temporal component, by 0.20 in F1 score. ViLT is the closest in performance to MEANT base, achieving an F1-score of 0.949. ViLT the most similar encoding structure to MEANT, which is one reason for the similar performance. The performance gains with MEANT emphasize the effectiveness of combining the SOTA transformer encoder architectures with temporal components.

MEANT-XL and MEANT-large are practically identical in performance, which indicates that the task is 'solved' with a model in the 170 million parameter range or so.

6.2 Stocknet results

Model	Acc%	F1	P	R
BERTweet	49.20	0.32	0.24	0.50
StockNet	57.53	0.57	0.58	0.57
TEANet	70.88	0.70	0.70	0.70
M-Tweet base	79.92	0.79	0.80	0.79
M-Tweet-large	80.17	0.80	0.80	0.80
M-Tweet-XL	**85.65**	**0.85**	**0.85**	**0.85**

Table 6: StockNet-dataset experiment results using Precision (P), Recall (R), F-1 scores and testing accuracy (Acc).

Looking at 6, MEANT-Tweets base and MEANT-Tweets large outperform all other models by a significant amount on the StockNet task. MEANT-tweet-XL outperformed TEANet, the previous SOTA on the StockNet dataset, by 15%. We ran our own implementation of the TEANet model on the task following their descriptions from the paper, as we could not find publicly available code. The original accuracy score reported in the paper was 65.16% (Zhang et al., 2022).

The importance of a temporal component for the StockNet task is clear. BERTweet, a typical encoder architecture without temporal support, performed abysmally. StockNet performed marginally better, but it is with the auxiliary temporal softmax mechanism in TEANet that the first true performance gain can be seen.

Clearly, the attention-based temporal mechanism in MEANT is the most performant for this problem. MEANT is able to extract meaningful relationships between the target day and the auxiliary trading days, in a way that allows for far more accurate binary price prediction then previously defined mechanisms. There are likely a few reasons for this. Models that depend on multi-head selt-attention (MSA) can be thought of as a low pass filters, meaning that they generally tend to flatten loss landscapes (Park and Kim, 2022). There are Tweets in the StockNet dataset that don't correlate to the buy signal, but because of the nature of the data collection, these are in the vast minority (Xu and Cohen, 2018). However, since we are also extracting trends that are dependent on the order of these Tweets in time, a succession of even a few outlier or irrelevant Tweets could be very damaging to the loss landscape of a more sensitive model. Our temporal attention mechanism is better able to handle the noise in the data. Furthermore, attention scales far better with parameter size, and our MEANT-XL model in particular dwarfs previous TEANet and StockNet in parameter size (Zhang et al., 2022; Xu and Cohen, 2018). Larger parameter spaces tend to lead to a more nuanced loss landscape (Fort and Jastrzebski, 2019; Fort and Scherlis, 2019; Park and Kim, 2022).

6.3 Albation Study

To examine the importance of the image and language modalities respectively, we also created two variations of the MEANT model, MEANT-vision and MEANT-language. MEANT-vision contained only the vision-encoder, while MEANT-Tweets used the language-encoder only. Both model still used the temporal attention head. These two variants were similarly fine-tuned and evaluated on the TempStock-small task 7.

Model	F1	P	R
MEANT-base	0.97	0.98	0.96
MEANT-large	0.99	**0.99**	0.99
MEANT-XL	**0.99**	0.98	**0.99**
M-Tweets	0.94	0.94	0.94
M-Tweets-large	0.95	0.95	0.95
M-Tweets-XL	0.95	0.95	0.95
M-vision	0.72	0.77	0.73
M-vision-large	0.74	0.74	0.74
M-vision-XL	0.77	0.76	0.78

Table 7: TempStock MEANT-variant Results, using Precision (P), Recall (R), and F-1 scores.

7 shows that MEANT large exhibited the best performance in F1, precision, and recall. What is perhaps more interesting about these results is examining the performance of MEANT-Tweet vs MEANT-large and MEANT-XL. The performance drop-off from MEANT-base to MEANT-Tweets-base is only about 0.03 in F1 score. Yet MEANT-vision-base exhibits a performance drop off of 0.25 from MEANT-base. These results indicate that the Twitter inputs contain features which are more indicative of momentum changes in the MACD indicator than the long-range graph inputs. This makes sense, as the graph images are sparsely populated (being mostly white space) and thus contain less information at face value. We are training our vision encoders to sort through a lot of blank noise to find the relevant information, which likely requires more rigorous pretraining schemes to realize the

true benefits of our long range information (Park and Kim, 2022; Dosovitskiy et al., 2020).

6.4 Discussion

Here, we outline considerations, trade-offs and design decisions we have made:

- **Dataset** To explore temporal information processing, we chose momentum buy signals in stock market data. We went with the MACD indicator because of its robustness, and correlation to strong positive returns against other indicators (Joshi, 2022; Chio, 2022). The serious drawback in this choice is in the infrequency of buy signals that occur. To alleviate the huge class imbalance, we decided to use the SMOTE algorithm to produce synthetic examples. We chose oversampling as a technique over under sampling, because of the relatively small size of our evaluation dataset. This method has some drawbacks. SMOTE might generate examples in areas where classes overlap or there is noise, away from more secure regions. This could result in the creation of instances that do not accurately reflect the characteristics of the minority class, potentially degrading the effectiveness of classification (Elreedy and Atiya, 2019; Teslenko et al., 2023). Furthermore, the precision of the instances produced by SMOTE can be affected by various factors, including the dataset's dimensionality, the training set's size, and the chosen number of nearest neighbors (Elreedy and Atiya, 2019; Teslenko et al., 2023; Grina et al., 2020). We gathered our stock price information from companies in the S&P 500. We chose this index because of its stability. However, as a result, we were unable to train our model on more extreme price patterns that are more common on obscure indexes (Goetzmann and Massa, 2003). Thus, in the case of extreme market events that result in periods of steep decline or rise would likely confuse the model.

- **MEANT** The MEANT encoder is built atop the Kosmos-1 encoder architecture, that uses interleaved LayerNorms (Vu et al., 2022). The authors thought this to lead to increased numeric stability (Huang et al., 2023), which in turn helps prevent the exploding gradient problem. However, the inclusion of so many layerNorms in each encoder in our models can lead to an increase in bias, which eventually can lead to a serious overfitting problem (Xu et al., 2019). We chose to go ahead with this risk, as previous architectures have shown the stability gains from the interleaved normalizations to allow for better scaling (Wang et al., 2022; Huang et al., 2023). MEANT was trained to identify buy signals, and reject everything else, instead of trying to classify price periods on a more nuanced scale. We chose this path for simplicity's sake. For practical use on financial data, we would likely need more levels of categorization.

7 Conclusion and Future Work

We introduced a multimodal encoder with a novel temporal component comprised entirely of self-attention. MEANT outperforms previous models on the StockNet benchmark by 15%, and proves to be the most performant model on our own TempStock benchmark. To our knowledge, MEANT-XL is the largest model to be applied to StockNet, and is the first multimodal model to contain an attention mechanism to deal with data over a lag period of days. MEANT combines the realms of language, vision, and time to produce SOTA results. In the future, we would like to test MEANT against some common multimodal benchmarks, such as Visual Question Answering (VQA) and Visual Commonsense Reasoning (VCR). We believe that the MEANT architecture has the potential to succeed on a wide variety of tasks. Furthermore, the image space that we trained MEANT on was limited. We would like to introduce more variation into our image inputs, to fully utilize the capabilities of that modality in our model.

8 Ethics Statement

Bias and Data Privacy: We acknowledge that there are biases in our study, including limiting our work to a specific time period, a small sample of securities and the general public, where we cannot verify they financial expertise in assessing markets. The data collected in this work will only be made available via Tweet IDs collected to protect X's users rights to remove, withdraw or delete their content. All datasets and Language Models are publicly available and were used under the license category that allows use for academic research.

Reproducibility: We make all of our code publicly available upon publication on Github, where we provide instructions to reproduce our results.

Use case: We strongly advise against the use of our proposed model and dataset for financial decision making, including automated or high frequency trading.

References

Alpha Vantage Inc. 2024. Alphavantage api. Retrieved February 2024.

G. Appel. 2005. *Technical Analysis: Power Tools for Active Investors*. Financial Times Prentice Hall books. Financial Times/Prentice Hall.

Dogu Araci. 2019. Finbert: Financial sentiment analysis with pre-trained language models.

Matin N. Ashtiani and Bijan Raahemi. 2023. News-based intelligent prediction of financial markets using text mining and machine learning: A systematic literature review. *Expert Systems with Applications*, 217:119509.

Kevin W. Bowyer, Nitesh V. Chawla, Lawrence O. Hall, and W. Philip Kegelmeyer. 2011. SMOTE: synthetic minority over-sampling technique. *CoRR*, abs/1106.1813.

Robert Goodell Brown. 1964. Smoothing, forecasting and prediction of discrete time series.

Leland Bybee, Bryan Kelly, and Yinan Su. 2023. Narrative Asset Pricing: Interpretable Systematic Risk Factors from News Text. *The Review of Financial Studies*, 36(12):4759–4787.

Pat Tong Chio. 2022. A comparative study of the MACD-base trading strategies: evidence from the US stock market. Papers 2206.12282, arXiv.org.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929.

Dina Elreedy and Amir F. Atiya. 2019. A comprehensive analysis of synthetic minority oversampling technique (smote) for handling class imbalance. *Information Sciences*.

Stanislav Fort and Stanislaw Jastrzebski. 2019. Large scale structure of neural network loss landscapes. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Stanislav Fort and Adam Scherlis. 2019. The goldilocks zone: towards better understanding of neural network loss landscapes. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence and Thirty-First Innovative Applications of Artificial Intelligence Conference and Ninth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

William N. Goetzmann and Massimo Massa. 2003. Index funds and stock market growth. *The Journal of Business*, 76(1):1–28.

Fares Grina, Zied Elouedi, and Eric Lefevre. 2020. *A Preprocessing Approach for Class-Imbalanced Data Using SMOTE and Belief Function Theory*.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415.

Guang-Bin Huang, Hongming Zhou, Xiaojian Ding, and Rui Zhang. 2012. Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):513–529.

Shaohan Huang, Li Dong, Wenhui Wang, Yaru Hao, Saksham Singhal, Shuming Ma, Tengchao Lv, Lei Cui, Owais Khan Mohammed, Barun Patra, Qiang Liu, Kriti Aggarwal, Zewen Chi, Johan Bjorck, Vishrav Chaudhary, Subhojit Som, Xia Song, and Furu Wei. 2023. Language is not all you need: Aligning perception with language models.

Yifu Huang, Kai Huang, Yang Wang, H. Zhang, Jihong Guan, and Shuigeng Zhou. 2016. Exploiting twitter moods to boost financial trend prediction based on deep network models. In *International Conference on Intelligent Computing*.

Dushyant Joshi. 2022. Use of moving average convergence divergence for predicting price movements. *International Research Journal of MMC*, 3:21–25.

JustAnotherArchivist. 2021. snscrape: A social networking service scraper in python. Version 0.34.

Wonjae Kim, Bokyung Son, and Ildoo Kim. 2021. Vilt: Vision-and-language transformer without convolution or region supervision. In *International Conference on Machine Learning*.

Qing Li, Jinghua Tan, Jun Wang, and Hsinchun Chen. 2021. A multimodal event-driven lstm model for stock prediction using online news. *IEEE Transactions on Knowledge and Data Engineering*, 33(10):3323–3337.

Paul Pu Liang, Yiwei Lyu, Xiang Fan, Zetian Wu, Yun Cheng, Jason Wu, Leslie Chen, Peter Wu, Michelle A. Lee, Yuke Zhu, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2021. Multibench: Multiscale benchmarks for multimodal representation learning. *CoRR*, abs/2107.07502.

Ilya Loshchilov and Frank Hutter. 2016. SGDR: stochastic gradient descent with restarts. *CoRR*, abs/1608.03983.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *CoRR*, abs/1908.02265.

Yuexin Mao, Wei Wei, Bing Wang, and Benyuan Liu. 2012. Correlating sp 500 stocks with twitter data. In *Proceedings of the First ACM International Workshop on Hot Topics on Interdisciplinary Social Networks Research*, HotSocial '12, page 69–72, New York, NY, USA. Association for Computing Machinery.

Marc-andre Mittermayer and Gerhard F. Knolmayer. 2006. Newscats: A news categorization and trading system. In *Sixth International Conference on Data Mining (ICDM'06)*, pages 1002–1007.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets.

OpenAI, :, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2023. Gpt-4 technical report.

Namuk Park and Songkuk Kim. 2022. How do vision transformers work?

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio.

2013. On the difficulty of training recurrent neural networks.

Yulong Pei, Amarachi Mbakwe, Akshat Gupta, Salwa Alamir, Hanxuan Lin, Xiaomo Liu, and Sameena Shah. 2022. TweetFinSent: A dataset of stock sentiments on Twitter. In *Proceedings of the Fourth Workshop on Financial Technology and Natural Language Processing (FinNLP)*, pages 37–47, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Sreelekshmy Selvin, R Vinayakumar, EA Gopalakrishnan, Vijay Krishna Menon, and KP Soman. 2017. Stock price prediction using lstm, rnn and cnn-sliding window model. In *2017 international conference on advances in computing, communications and informatics (icacci)*, pages 1643–1647. IEEE.

Thársis Tuani Pinto Souza, Olga Kolchyna, Philip C. Treleaven, and Tomaso Aste. 2015. Twitter sentiment analysis applied to finance: A case study in the retail industry. *CoRR*, abs/1507.00784.

Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced transformer with rotary position embedding. *CoRR*, abs/2104.09864.

Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. 2019. VL-BERT: pre-training of generic visual-linguistic representations. *CoRR*, abs/1908.08530.

Weijie Su, Xizhou Zhu, Chenxin Tao, Lewei Lu, Bin Li, Gao Huang, Yu Qiao, Xiaogang Wang, Jie Zhou, and Jifeng Dai. 2023. Towards all-in-one pre-training via maximizing multi-modal mutual information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15888–15899.

Tong Sun, Jia Wang, Pengfei Zhang, Yu Cao, Benyuan Liu, and Degang Wang. 2017. Predicting stock price returns using microblog sentiment for chinese stock market. In *2017 3rd International Conference on Big Data Computing and Communications (BIGCOM)*, pages 87–96.

Yutao Sun, Li Dong, Barun Patra, Shuming Ma, Shaohan Huang, Alon Benhaim, Vishrav Chaudhary, Xia Song, and Furu Wei. 2022. A length-extrapolatable transformer.

Denys Teslenko, Anna Sorokina, Artem Khovrat, Nural Huliiev, and Valentyna Kyriy. 2023. Comparison of dataset oversampling algorithms and their applicability to the categorization problem. *Innovative Technologies and Scientific Solutions for Industries*, pages 161–171.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Doan Nam Long Vu, Nafise Sadat Moosavi, and Steffen Eger. 2022. Layer or representation space: What makes BERT-based evaluation metrics robust? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3401–3411, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Hongyu Wang, Shuming Ma, Shaohan Huang, Li Dong, Wenhui Wang, Zhiliang Peng, Yu Wu, Payal Bajaj, Saksham Singhal, Alon Benhaim, Barun Patra, Zhun Liu, Vishrav Chaudhary, Xia Song, and Furu Wei. 2022. Foundation transformers.

Yi-Fan Wang. 2003. On-demand forecasting of stock prices using a real-time predictor. *Knowledge and Data Engineering, IEEE Transactions on*, 15:1033–1037.

Huizhe Wu, Wei Zhang, Weiwei Shen, and Jun Wang. 2018. Hybrid deep sequential modeling for social text-driven stock prediction. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, CIKM '18, page 1627–1630, New York, NY, USA. Association for Computing Machinery.

Jingjing Xu, Xu Sun, Zhiyuan Zhang, Guangxiang Zhao, and Junyang Lin. 2019. Understanding and improving layer normalization. *CoRR*, abs/1911.07013.

Yumo Xu and Shay B. Cohen. 2018. Stock movement prediction from tweets and historical prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1979, Melbourne, Australia. Association for Computational Linguistics.

Biao Zhang and Rico Sennrich. 2019. Root mean square layer normalization. *CoRR*, abs/1910.07467.

Qiuyue Zhang, Chao Qin, Yunfeng Zhang, Fangxun Bao, Caiming Zhang, and Peide Liu. 2022. Transformer-based attention network for stock movement prediction. *Expert Systems with Applications*, 202:117239.

Yi Zuo, Masaaki Harada, Takao Mizuno, and Eisuke Kita. 2012. Bayesian network based prediction algorithm of stock price return. In *Intelligent Decision Technologies*, pages 397–406, Berlin, Heidelberg. Springer Berlin Heidelberg.